

UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR



Grado en Ingeniería Informática

TRABAJO FIN DE GRADO

**Realización de una aplicación de mensajería instantánea
basada en Android para firmas**

Alberto Riaño González

Tutor: Gonzalo Martínez Muñoz

Julio 2015

Resumen

Los teléfonos inteligentes han cobrado un papel muy importante en la sociedad. Debido al auge de estos dispositivos, la mayoría de la población hace un uso diario de los servicios que ofrecen, siendo Android el sistema operativo más presente en el mercado.

Actualmente, la comunicación entre alumnos y profesores a nivel universitario reside, principalmente, en el correo electrónico y plataformas educativas.

Aunque eficientes, son alternativas desarrolladas en un inicio para otros sistemas operativos. Por ello, el objetivo principal del proyecto se basa en ofrecer una nueva alternativa de comunicación para alumnos y profesores, más especializada y dirigida a los actuales avances tecnológicos. De este modo, se ha desarrollado una aplicación móvil para Android que permita alcanzar dicho objetivo, ofreciendo una opción más rápida y efectiva de comunicación.

La aplicación creada está estructurada por asignaturas en las que los usuarios pueden tratar diferentes temas a partir de un servicio de mensajería instantánea. Desde el punto de vista docente, se podrán agilizar los procesos de notificaciones masivas a alumnos, como pueden ser: alertar de una posible modificación de clase, la publicación de calificaciones o material didáctico en la plataforma educativa. Además, alumnos y profesores podrán establecer tutorías e interactuar entre sí mediante mensajería privada.

A lo largo del documento se presenta la aplicación desarrollada, exponiendo las fases por las que ha pasado el proyecto y las posibilidades que ofrece de cara a un futuro.

Palabras clave

Android, aplicación, universidad, chat, asignatura, alumno, profesor, tutoría, notificación

Abstract

Smartphones play a very important role in our society. Due to the rise of these devices, the main part of the population uses daily their services, with Android as the more popular operating system in market.

Currently, communication between students and teachers at university is carried out through email and educational platforms.

Although efficient, alternatives are developed initially thinking of other operating systems. Therefore, the aim of the project is to offer a new alternative communication for students and teachers, more specialized and using current technological advances. For this reason, it has been developed an Android mobile application which would achieve this goal by offering a faster and more effective communication option.

The application created is structured by subjects in which users can try different themes through an instant messaging service. From the educational point of view, teachers can streamline the process of mass notifications to students, warning of a possible change in class, posting grades or teaching material in the educational platform. In addition, students and teachers can establish tutorials and interact with each other by private messaging.

Throughout the document the developed application is presented, exposing the phases of development of the project and the possibilities that it offers regarding the future.

Key words

Android, application, university, chat, subject, student, professor, tutorial, notification

ÍNDICE DE CONTENIDOS

1. Introducción	1
1.1 Motivación y definición del problema	1
1.2 Objetivos del proyecto.....	2
1.3 Estructura del documento	2
2. Estado del arte	5
2.1 Android	5
2.1.1 Arquitectura de Android	6
2.1.2 Versiones del sistema operativo	6
2.2 Enfoques existentes	7
2.2.1 Ámbito universitario	7
2.2.2 Servicio de mensajería instantánea	8
3. Análisis	11
3.1 Especificación de requisitos	11
3.1.1 Requisitos funcionales	11
3.1.2 Requisitos no funcionales	14
4. Diseño.....	15
4.1 Arquitectura de la aplicación	15
4.2 Cliente: Aplicación móvil	16
4.2.1 Prototipo de aplicación	16
4.2.2 Estilo de la aplicación	17
4.2.3 Base de datos local	17
4.3 Servidor	18
4.3.1 Base de datos	18
4.3.2 PHP	19
5. Implementación	21
5.1 Tecnologías utilizadas	21
5.1.1 Eclipse	21
5.1.2 PHP	21
5.1.3 PostgreSQL	21
5.1.4 Volley	21

5.1.5 Imágenes 9-path	22
5.2 Google Cloud Messaging	23
5.3 Flujo de la aplicación	24
5.4 Componentes de la interfaz gráfica	25
5.4.1 Login y pantalla de inicio	25
5.4.2 Menú desplegable	26
5.4.3 Menú asignatura	27
5.4.4 Interacción con temas de la asignatura	28
5.4.5 Establecimiento de tutoría	29
5.4.6 Mensajería privada	30
6. Pruebas de la aplicación	33
6.1 Pruebas en el cliente	33
6.2 Pruebas en el servidor	37
7. Conclusiones y trabajo futuro	39
7.1 Conclusiones	39
7.2 Trabajo futuro	40
7.2.1 Vinculación directa con Moodle o Sigma	40
7.2.2 Calendario personal para el alumno	40
7.2.3 Relación con servicios internos de la universidad	41
Referencias	42

ÍNDICE DE TABLAS

Tabla 1: Ficheros PHP servidor	19
Tabla 2: Pruebas cliente: Inicio y cierre sesión	33
Tabla 3: Pruebas cliente: Interacción con temas de asignaturas	34
Tabla 4: Pruebas cliente: Intercambio de mensajería privada	35
Tabla 5: Pruebas cliente: Tutoría y avisos de asignaturas	36
Tabla 6: Pruebas servidor	37

ÍNDICE DE ILUSTRACIONES

Ilustración 1: Capas de la arquitectura Android	5
Ilustración 2: Versiones del Sistema Operativo Android	6
Ilustración 3: Gráfico del reparto de versiones Android 2015	7
Ilustración 4: Capturas de pantalla aplicación UC3M Aula Global	7
Ilustración 5: Capturas de pantalla aplicación WhatsApp	8
Ilustración 6: Logotipos aplicaciones de mensajería instantánea	9
Ilustración 7: Arquitectura de la aplicación	15
Ilustración 8: Maquetas	16
Ilustración 9: Diagrama Entidad-Relación base de datos servidor	18
Ilustración 10: Imágenes Nine-Path	22
Ilustración 11: Arquitectura GCM	23
Ilustración 12: Diagrama de transición de pantalla de la aplicación	24
Ilustración 13: Conjunto de imágenes login y pantalla de inicio	25
Ilustración 14: Conjunto de imágenes menú desplegable	26
Ilustración 15: Conjunto de imágenes interacción con temas	27
Ilustración 16: Conjunto de imágenes ver profesores y lanzar avisos	27
Ilustración 17: Conjunto de imágenes interacción con temas	28
Ilustración 18: Notificación nuevo mensaje en tema	28
Ilustración 19: Conjunto de imágenes establecimiento de tutoría	29
Ilustración 20: Denegación de tutoría	30
Ilustración 21: Conjunto de imágenes mensajería privada	30
Ilustración 22: Notificación nuevo mensaje privado	31

Introducción

En este apartado se aporta una breve descripción de los objetivos y la motivación principal para la realización de una aplicación basada en Android para asignaturas, así como la estructura que seguirá el presente trabajo fin de grado.

1.1 Motivación y definición del problema

El proceso de comunicación se ha ido modificando a lo largo del tiempo, el correo electrónico se convirtió en un gran instrumento para transmitir mensajes y alertas urgentes. Sin embargo, aunque la frecuencia media con la que un usuario mira su correo es muy variada según qué personas, el uso del mismo ha descendido con la aparición de los teléfonos inteligentes.

Esta frecuencia de seguimiento contrasta drásticamente con la que es capaz de ofrecer un dispositivo móvil. Actualmente, según las cifras facilitadas por el Instituto Nacional de Estadística (INE, en lo sucesivo), el 88,2% de la población dispone de un dispositivo móvil [1]. Este porcentaje se dispara a un 98,8% en personas con rangos de edad de entre 18 y 30 años [1], pues es en este intervalo donde se encuentra la inmensa mayoría de estudiantes universitarios.

De este modo y, confrontando estas estadísticas, resultaría complicado encontrar un usuario que consultase más frecuentemente su correo electrónico que su dispositivo móvil.

Centrándonos en el ámbito universitario, el proceso de comunicación entre profesores y alumnos se produce la mayoría de veces a través de escritos enviados desde diversas plataformas, como puede ser el correo electrónico o Moodle, cuyo acceso por alumnos y profesores se puede producir de manera puntual.

Las notificaciones por parte de los profesores pueden ser muy diversas, desde publicación de notas, subida de cierto material hasta notificaciones de carácter más urgente como pudiera ser la cancelación de una clase por incomparecencia de un profesor, el cambio de aula debido a que no se pudiera impartir la clase en el lugar previsto o cualquier tipo de variación en un examen próximo. Además, estas notificaciones por parte del profesor a los alumnos suelen realizarse de forma masiva.

Por todo esto, nace la motivación de crear PRALUAM (PRofesores y ALumnos de la UAM), proyecto que genera numerosas ventajas en el proceso comunicativo entre profesores y alumnos, mejorando la fluidez de las mismas.

PRALUAM permitirá que los alumnos que así lo deseen reciban notificaciones móviles con los avisos que los profesores pudieran mandar a través del correo electrónico o la plataforma educativa universitaria.

1.2 Objetivos del proyecto

El objetivo principal es mejorar el proceso de comunicación entre profesores y alumnos a través de la implementación de una aplicación para dispositivos móviles que ofrezca nuevas ventajas en el ámbito educativo universitario.

Para poder llevar a cabo este objetivo, la aplicación desarrollada brinda diversas posibilidades comunicativas, entre las que destacan las siguientes:

- **Comunicación bajo mensajería instantánea** en temas clasificados por asignaturas. De este modo, existirá un tema general, así como la posibilidad de crear nuevos temas tanto por alumnos como profesores con variados fines tales como aclarar dudas, informar de nuevas fuentes de información, solicitar ayuda en un determinado ejercicio, etc. Los usuarios a través de la aplicación seleccionarán sobre qué temas desean ser notificados.
- **Intercambio de mensajería privada.**
- **Petición y establecimiento de una tutoría.** Este servicio ofrecerá la posibilidad a un alumno de solicitar una tutoría incluyendo su disponibilidad así como sus opciones de máxima preferencia. El profesor, al recibir la solicitud, puede bien rechazarla o aceptar alguna de las propuestas. Si se confirma la tutoría, el sistema almacena en el calendario del dispositivo móvil un evento en el que se fija la hora y fecha de la reunión, alertándoles el mismo día de la cita a modo de recordatorio.
- **Notificaciones masivas de los profesores a sus alumnos**, informándoles de las siguientes novedades disponibles en la plataforma educativa:
 - Calificaciones publicadas
 - Contenido didáctico subido a la plataforma educativa
 - Cancelación/modificación de una clase

1.3 Estructura del documento

El documento se distribuye en siete capítulos. En estos capítulos se trata una parte importante del análisis, diseño, implementación o evaluación de la aplicación desarrollada.

- En el **primer capítulo** se lleva a cabo una introducción al trabajo realizado, la motivación del mismo y los objetivos que se pretenden alcanzar.

- El **segundo capítulo** está dedicado al estado del arte. En él se trata el sistema operativo Android y enfoques ya existentes de aplicaciones con fines similares al presente proyecto.
- El **tercer capítulo** trata la fase de análisis del proyecto detallando la especificación de requisitos que ha de cumplir la aplicación.
- En el **cuarto capítulo** se explica la arquitectura de la aplicación así como su diseño, diferenciando entre la parte del cliente y la del servidor.
- El **quinto capítulo** se dedica a la implementación de la aplicación. Se presenta la aplicación desarrollada y se detallan las tecnologías utilizadas para llevarla a cabo.
- En el **sexto capítulo** se enumeran las pruebas realizadas una vez superadas las fases anteriores.
- En el **séptimo capítulo** se exponen las conclusiones finales y se estudian una serie de variantes ante una posible continuación del proyecto.

Estado del arte

2.1 Android

Android es un sistema operativo basado en el núcleo de Linux diseñado principalmente para dispositivos móviles con pantalla táctil, tanto smartphones como tablets. En sus orígenes fue desarrollado por Android Inc., empresa que Google respaldó económicamente para posteriormente comprarla en 2005.

Fue presentado en 2007 para empezar a comercializarse en octubre de 2008 a través de HTC Dream, primer dispositivo móvil que funcionaba bajo el sistema operativo Android. Uno de los aspectos más novedosos que ofrecía Android era su orientación a la multiplataforma, algo revolucionario teniendo en cuenta que se venía asociando un sistema operativo a un único dispositivo.

Rápidamente Android fue expandiéndose hasta lograr, ya en 2010, implantar su sistema operativo en más de la mitad de los dispositivos móviles.

2.1.1 Arquitectura de Android

Como muestra la ilustración 1, Android se divide en 5 capas principales en la que cada una de ellas utiliza elementos de su capa inferior para llevar a cabo sus funciones.

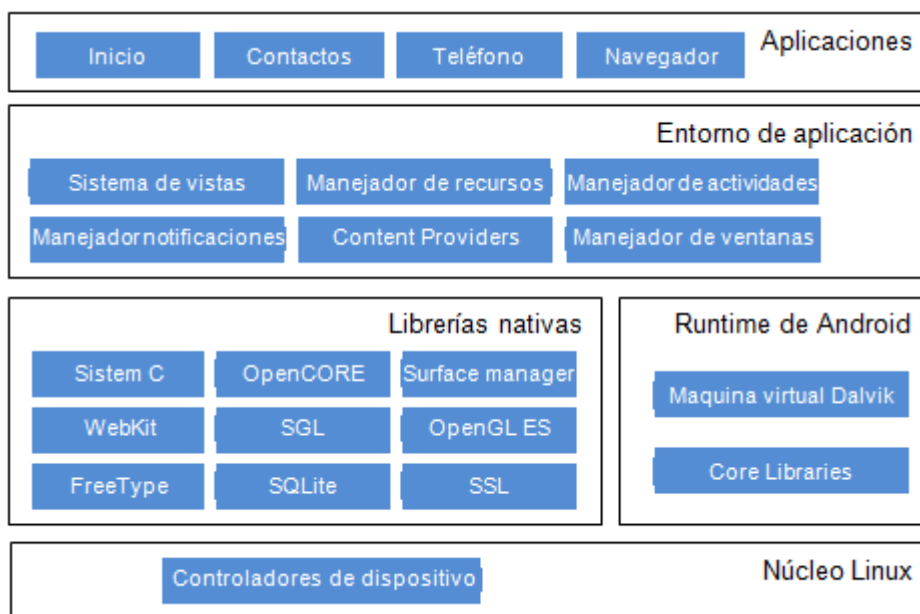


Ilustración 1: Capas de la arquitectura Android [2]

Núcleo de Linux: Es la primera capa del sistema Android. Tiene dependencia de Linux para los servicios base del sistema (seguridad, gestión de memoria, pila de red, etc). A su vez, actúa como una capa de abstracción entre el hardware y software del dispositivo.

Runtime de Android: Se encarga de llevar a cabo la transformación de código de bytes de la aplicación en instrucciones nativas que serán ejecutadas por el entorno de ejecución del dispositivo. Para llevar a cabo la transformación accede a recursos de capas inferiores a través de la máquina virtual Dalvik. No obstante, a partir de Android 5.0 lo lleva a cabo a partir de ART, máquina que reemplaza a Dalvik.

Librerías nativas: Esta capa incluye un conjunto de librerías escritas en C/C++ y compiladas para la arquitectura hardware concreta del dispositivo móvil. Estas librerías proporcionan a Android la mayor parte de sus capacidades más características.

Entorno de aplicación: Representa el conjunto de herramientas de desarrollo de aplicaciones. Todas las aplicaciones desarrolladas para Android hacen uso del mismo conjunto de API entre las que destacan activity manager, window manager, view system, location manager, etc

Aplicaciones: Las aplicaciones conforman la última capa utilizando los servicios, las APIs y las librerías de los niveles inferiores. Están codificadas en lenguaje Java y engloban aquellas incluidas por defecto de Android y las creadas por el usuario.

2.1.2 Versiones del sistema operativo

Android creó su primera versión en septiembre de 2008, aunque no fue hasta abril de 2009 cuando se lanzó el sistema operativo con Android 1.5 [3]. Desde entonces no ha parado de crecer lanzando nuevas versiones que incorporan nuevas funcionalidades o mejoran las existentes.

Cada una de estas versiones recibe el nombre de postres en inglés y siguen un orden alfabético como muestra la ilustración 2:



Ilustración 2: Versiones del Sistema Operativo Android [4]

Actualmente, estas versiones conviven en los dispositivos móviles, debido a que muchos de ellos no se han actualizado a las nuevas versiones. En la ilustración 3 puede observarse el porcentaje de terminales que usan las diferentes versiones.

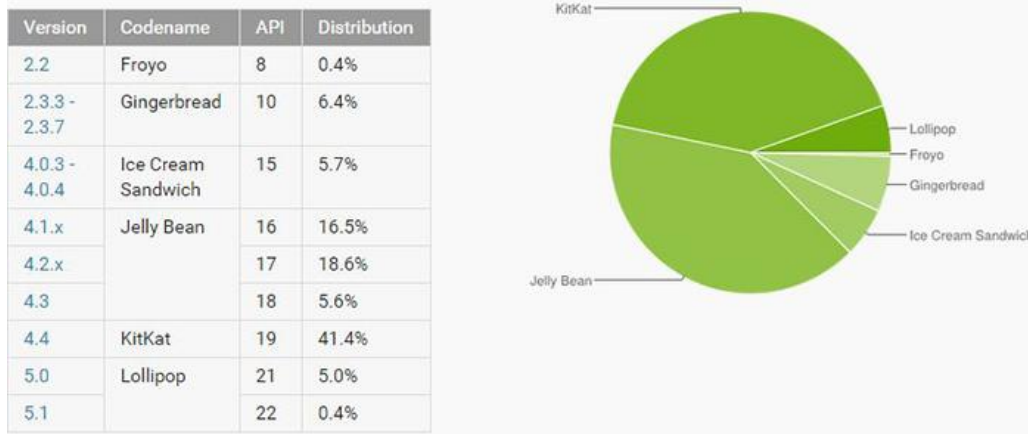


Ilustración 3: Gráfico del reparto de versiones Android 2015 [5]

2.2 Enfoques existentes

En este apartado analizaremos enfoques existentes de aplicaciones destinadas tanto al ámbito universitario como a ofrecer un servicio de mensajería instantánea.

2.2.1 Ámbito universitario

A día de hoy, la mayoría de aplicaciones móviles destinadas al entorno educativo de nivel superior están relacionadas con la consulta del expediente académico de los estudiantes.

La Universidad Carlos III de Madrid [6] dispone de la aplicación UC3M Aula Global (ver capturas en la ilustración 4).

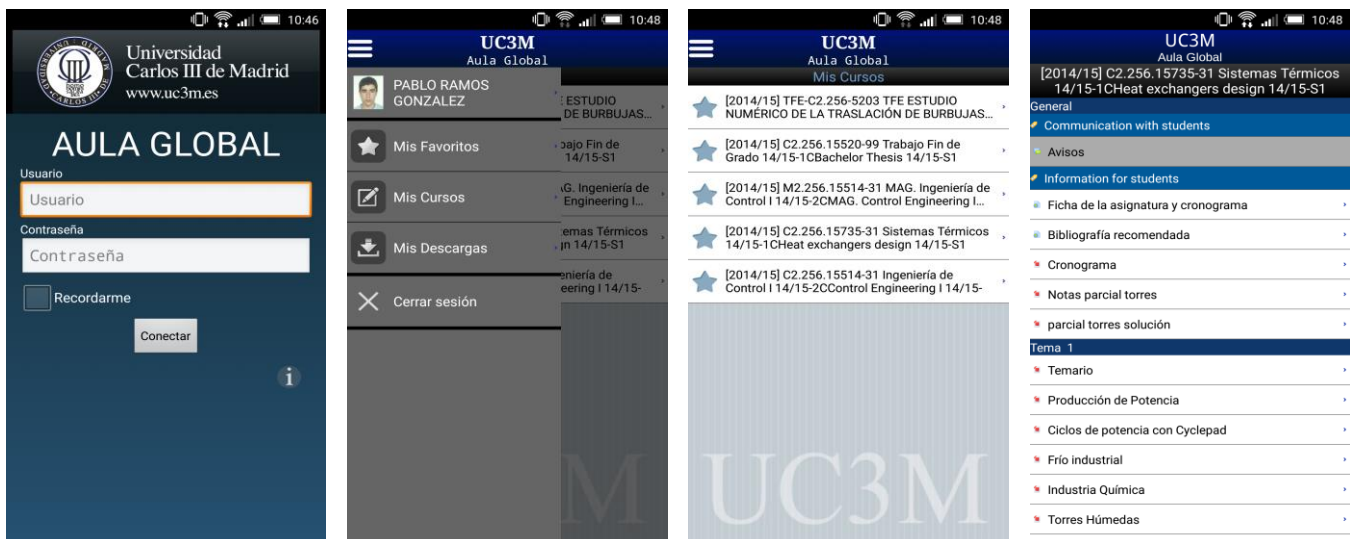


Ilustración 4: Capturas de pantalla aplicación UC3M Aula Global

La aplicación, además de dar acceso al expediente del alumno, proporciona una vinculación directa con el Moodle (o Aula global) de la Universidad, cargando el perfil del estudiante con sus asignaturas y permitiendo ver el contenido subido en cada una de ellas. La principal ventaja que ofrece es que dicho contenido se puede descargar en el dispositivo móvil desde la propia app.

2.2.2 Servicio de mensajería instantánea

En la actualidad, las aplicaciones de mensajería instantánea imperan como medio de comunicación. En los últimos años, este tipo de mensajería, se ha impuesto al servicio SMS que ofrecen las compañías móviles. Esto se ha propiciado por ser un servicio de pago que oferta pocas funcionalidades respecto a las aplicaciones de mensajería instantánea. Además, cuenta con restricciones comunicativas como la extensión del mensaje, limitada por el número de caracteres.

Sin embargo, actualmente, existen diversas aplicaciones gratuitas o de bajo coste que permiten comunicarse con multitud de funcionalidades y sin limitaciones. Estas permiten la comunicación no solo a través de mensajes de texto, sino también posibilitando el intercambio de audio, vídeos, imágenes, etc.

De entre la multitud de aplicaciones desarrolladas desde la aparición del Smartphone, la más importante ha sido WhatsApp [7]. Esta se fundó en el año 2009 creciendo vertiginosamente hasta alcanzar los 700 millones de usuarios en 2015. En 2014, Facebook adquirió WhatsApp por 19 millones de dólares. Esta aplicación de mensajería instantánea brinda, como se ve en la ilustración 5, a través de una atractiva interfaz, la posibilidad de chatear, intercambiar contenido multimedia y otras funcionalidades con todos los contactos del usuario que dispongan de la aplicación.

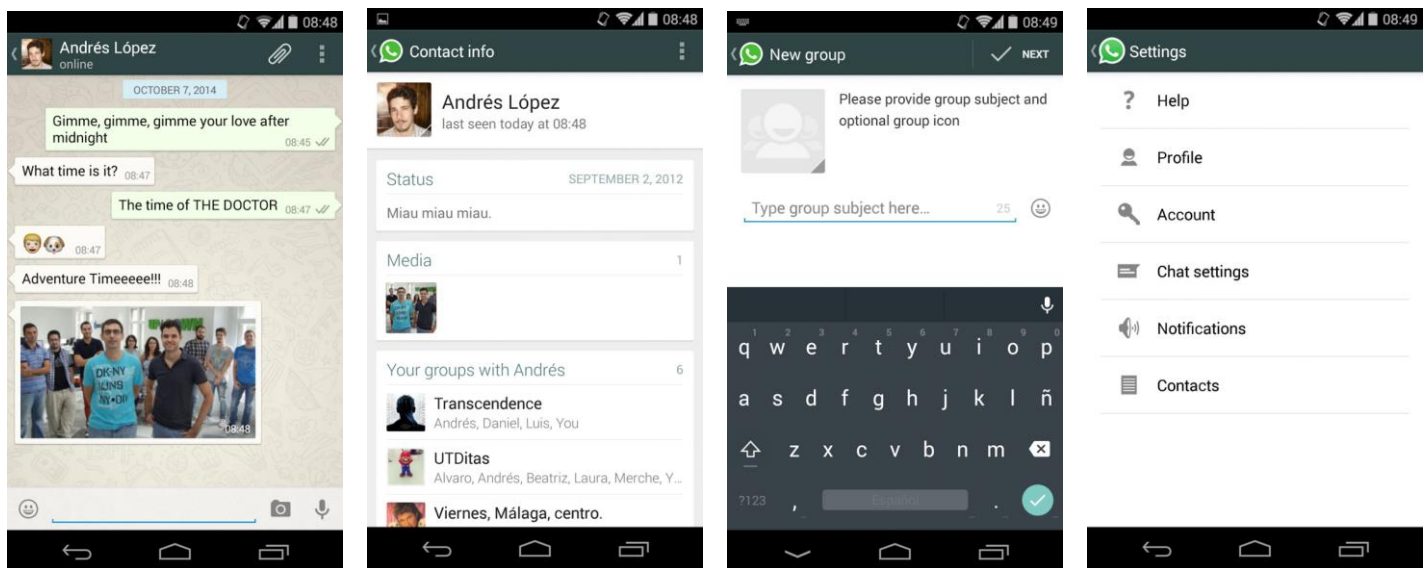


Ilustración 5: Capturas de pantalla aplicación WhatsApp [7]

Otras aplicaciones similares de mensajería instantánea son Telegram, Line o Viber. El secreto del éxito de alguna de estas reside en las diferencias que incluyen respecto al resto con el fin de diferenciarse. Por ejemplo, Telegram ofrece mensajería cifrada; Viber llamadas y video-llamadas; Line llamadas vía IP y una amplia gama de emoticonos.



Ilustración 6: Logotipos aplicaciones de mensajería instantánea

3.1 Especificación de requisitos

El catálogo de requisitos es la especificación del comportamiento que se espera de un proyecto software. A partir de un estudio de aplicaciones que ofrecen servicios similares, se predefinen una serie de requisitos de carácter indispensable para el proyecto. Estos se diferencian en funcionales y no funcionales.

3.1.1 Requisitos funcionales

Los requisitos funcionales describen las interacciones que el usuario puede tener con el software.

RF1: Identificación

El sistema debe validar el proceso de identificación de un usuario en la aplicación:

- El usuario se identificará a partir de su nombre de usuario y contraseña.
- El sistema se encargará de permitir o denegar el acceso a la aplicación. Si el acceso no se ha materializado, se mostrará un mensaje indicando el error producido (fallo en la introducción de datos o error en la conexión).

Cuando la identificación sea correcta, se mostrará un mensaje de bienvenida y se redirigirá al usuario a la pantalla principal. Además, el sistema vinculará el dispositivo móvil desde el que se realiza la identificación con el usuario en cuestión para posibilitar futuras notificaciones.

RF2: Cierre de sesión

El sistema ha de permitir al usuario cerrar sesión:

- Se pedirá confirmación del cierre de la sesión por parte de la aplicación. De confirmarse, la aplicación se cerrará y el usuario tendrá que pasar de nuevo por la fase de identificación si desea volver a cargar su perfil.

RF3: Notificaciones

El usuario podrá en todo momento configurar las notificaciones seleccionando así sobre qué cosas desea ser notificado. Esto engloba:

- Notificaciones personales
 - Mensajería privada
 - Petición/confirmación de tutoría
- Notificaciones del foro
 - Nuevas respuestas en temas. El usuario puede elegir entre:
 - Todos los temas
 - Sólo los que ha participado
 - Únicamente los que ha iniciado
 - Ninguno

Además, si el usuario es un alumno tendrá también la opción de elegir si desea ser notificado de nuevos cambios en sus asignaturas:

- Notificaciones en asignaturas
 - Publicación de notas
 - Subida de contenido didáctico
 - Cambio/Cancelación de clase

Dichas configuraciones se almacenarán en las preferencias del dispositivo móvil ofreciendo así persistencia a las opciones elegidas.

RF4: Mensajería Privada

La aplicación ofrecerá la posibilidad a todos los usuarios de intercambiar mensajes con carácter privado.

- Visitando el perfil de un usuario aparece la opción de hacerle destinatario de un nuevo mensaje privado.
- Se ofrecerá la posibilidad de contestar un mensaje privado desde la propia bandeja de entrada sin crear una pantalla adicional.
- El destinatario, si así lo tiene configurado, será notificado de la llegada de un nuevo mensaje privado, mostrándole el contenido del mismo y la posibilidad de responder al abrir la notificación.

RF5: Foro de una asignatura

Cada asignatura dispondrá de un foro en el que se almacenarán los temas existentes. El usuario podrá:

RF5.1 Crear un tema

- Se solicitará al usuario que asigne un título al tema y escriba el primer comentario en él. Si no se cumplen estos dos indicativos, la aplicación mandará un mensaje de error avisándolo.

- Si el tema se ha creado correctamente, aparecerá en el primer lugar del listado y estará disponible para todos los usuarios que tengan esa asignatura.

RF5.2 Interactuar con un tema creado

- Se mostrarán los mensajes del tema seleccionado en orden cronológico. Cada mensaje vendrá detallado con el nombre de su creador, fecha de envío y contenido. En el caso de pulsar prolongadamente sobre alguno de los mensajes ajenos, la aplicación dará la opción de visitar el perfil del usuario.
- La aplicación almacenará en la base de datos del móvil los mensajes ya leídos para cargar en futuras entradas al tema únicamente mensajes nuevos.
- El usuario podrá escribir un nuevo mensaje a través de la tecla “enviar”. De manera inmediata se notificará a todos los usuarios que así lo tengan configurado. Los usuarios que tuvieran abierto el tema en el momento de la recepción, verán el nuevo mensaje de forma instantánea en la pantalla de su dispositivo móvil.

RF6: Establecimiento de una tutoría

La aplicación permitirá establecer una tutoría entre el alumno y el profesor.

- El alumno, visitando el perfil de un profesor, tendrá la opción de solicitarle una tutoría. En la petición se incluirán tantas opciones como se desee, pudiendo indicar algunas de ellas como preferentes. Con esto, el alumno muestra su disponibilidad al profesor.
- El profesor al recibir la notificación con la solicitud de tutoría por parte del alumno visualiza las opciones que éste le oferta. De este modo, tendrá dos opciones:
 - Si ninguna de las opciones recibidas se adecua a su calendario, el profesor rechazará la tutoría y la aplicación avisará al alumno para que, si lo desea, formule una nueva petición con un rango de opciones diferentes.
 - Por otro lado, el profesor puede aceptar la tutoría clickando sobre cualquiera de las propuestas, apareciendo así un último aviso al profesor preguntándole si desea establecer definitivamente la tutoría en la fecha indicada. De confirmarse, el sistema ofrece la posibilidad de guardar el evento en el dispositivo móvil avisando el mismo día de la tutoría para recordarlo. Del mismo modo, se envía una nueva notificación al alumno informándole de que el profesor ha accedido a su solicitud y de la misma forma, almacena en el dispositivo móvil el día y hora de la reunión.

RF7: Avisos de asignaturas

La aplicación ofrecerá la posibilidad de que un profesor notifique a sus alumnos la nueva información relacionada con su asignatura. A destacar:

- Calificaciones publicadas
- Contenido didáctico subido a la plataforma educativa
- Cancelación/modificación de una clase

El sistema dará la opción de añadir simultáneamente un mensaje aclaratorio como, por ejemplo, indicando a qué corresponden las notas subidas o que la clase se traslada al aula 7. Una vez sea lanzado el aviso por parte del profesor, todos los alumnos que así lo tengan configurado recibirán de manera instantánea en su dispositivo móvil dicha información.

3.1.2 Requisitos No funcionales

Los requisitos de carácter no funcional especifican criterios que valoran operaciones del sistema sin tener en cuenta su comportamiento.

RNF1: Usabilidad

- La aplicación ofrecerá usabilidad en todo dispositivo móvil que funcione bajo el sistema operativo Android.
- La aplicación estará soportada para versiones superiores a 3.2 Honeycomb.
- Será necesario disponer de una conexión estable a internet, ya sea mediante Wifi o datos móviles.

RNF2: Interfaz

- La aplicación brindará una interfaz sencilla e intuitiva, facilitando el uso de la misma por los usuarios.
- Se ofrecerá un diseño adaptable a los distintos tamaños y densidades de pantalla, siendo practicable en cualquier dispositivo móvil.

RNF3: Seguridad

- Para hacer uso de la aplicación será obligatoria su autenticación.
- En futuras sesiones el usuario no tendrá que volver a loguearse iniciándose automáticamente, salvo que éste cierre sesión.
- Las contraseñas de los usuarios se cifran con MD5 [8].

RNF4: Rendimiento

- Se esperan tiempos de respuesta inferiores a dos segundos en peticiones al servidor y por debajo de un segundo en consultas a la propia base de datos del dispositivo móvil.
- Control de errores con el fin de garantizar la estabilidad de la aplicación evitando un cierre inesperado de la misma.
- La aplicación no necesita estar en segundo plano para recibir notificaciones debido a la incorporación de Google Cloud Messaging (GCM) [9].
- Los datos almacenados por la aplicación serán únicamente limitados por la propia capacidad de almacenamiento del dispositivo.

RNF5: Mantenimiento

- Se ofrecerá mantenimiento a la aplicación adaptándola a futuras versiones del sistema operativo Android.

4.1 Arquitectura de la aplicación

Esta sección describe la arquitectura llevada a cabo para la implementación de la aplicación móvil.

La unión de los distintos módulos da lugar a un esquema modular compacto. Este diseño hace posible la actualización o reemplazamiento de componentes evitando que el resto de la aplicación se vea afectada.

El diseño de la arquitectura global es el mostrado en la ilustración 7:

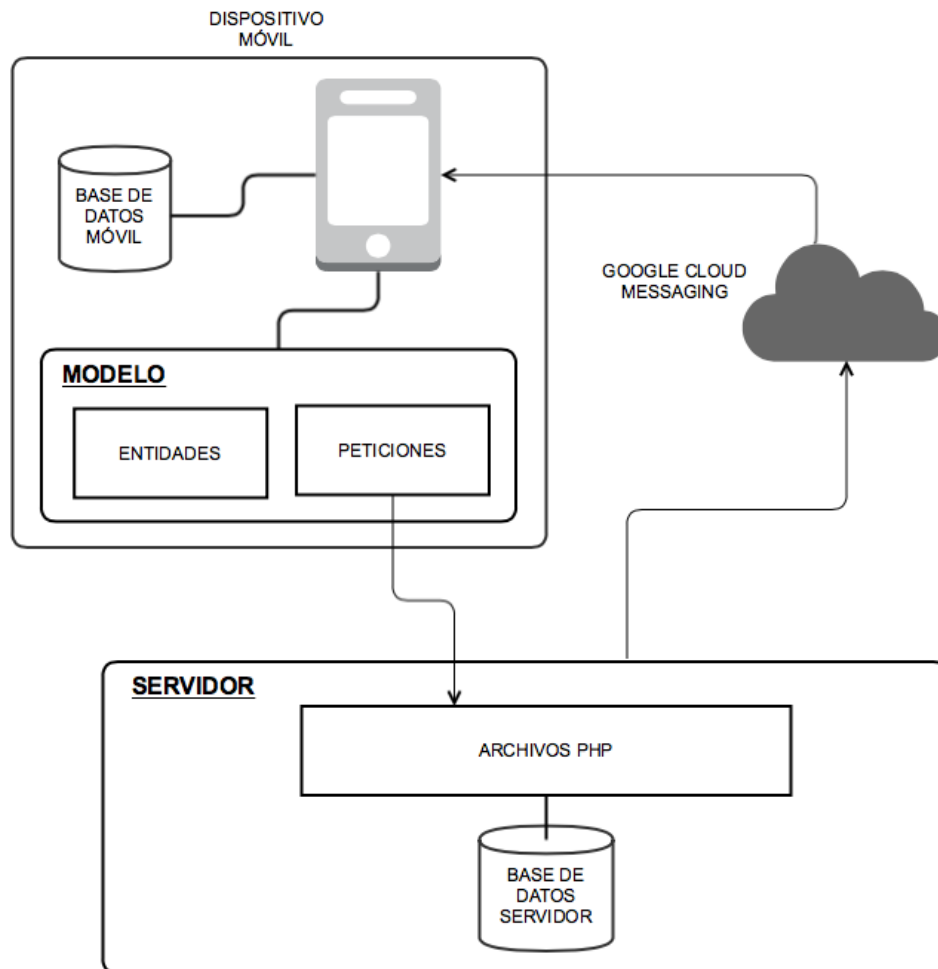


Ilustración 7: Arquitectura de la aplicación

La arquitectura, basada en el modelo vista controlador, está formada por 3 módulos principales: cliente, servidor y GCM. Los archivos PHP obtienen información a partir de la base de datos para formar la API de acceso del servidor. El dispositivo móvil, que accede a las entidades del modelo y formula peticiones al servidor, toma el papel de cliente. GCM es un módulo externo en el que se apoyan ambos para llevar a cabo las notificaciones push.

4.2 Cliente: Aplicación móvil

La aplicación móvil hace las funciones de cliente ya que se encarga de realizar las peticiones al servidor y procesarlas para, a través del dispositivo móvil, ofrecer al usuario el resultado deseado.

El aspecto visual de la aplicación tiene sus orígenes en el prototipo diseñado en un primer momento y el estilo seleccionado. Entraremos en detalle en los siguientes puntos.

La aplicación, además, cuenta con su base de datos propia como veremos más adelante.

4.2.1 Prototipo de la aplicación

En el origen del proyecto, a partir de unas maquetas, se diseñó un prototipo de aplicación que cumpliera con los requisitos funcionales enumerados en el apartado 3.1.1. Algunas de estas maquetas las podemos ver en la ilustración 8.

No obstante, como veremos en el capítulo de implementación, el prototipo inicial dista notablemente del resultado final, ya que se han ido añadiendo funcionalidades y mejoras visuales en el desarrollo de la misma.



Ilustración 8: Maquetas

4.2.2 Estilo de la aplicación

El aspecto visual de una aplicación es muy importante. Por este motivo, se ha procurado dar una interfaz amigable y personificada con el propósito de establecer una identidad que cause buen impacto al usuario final.

Para optimizar el *look&feel* se ha creado una serie de estilos y temas en el directorio *res/values/styles.xml* que ofrece el desarrollo de Android para que las vistas de la aplicación puedan hacer uso de ellos.

Los estilos permiten modificar el aspecto de una vista definiendo propiedades como altura, padding, fondo, color o tamaño de fuente, etc. Un estilo se define en un archivo XML separado de la definición del layout. Esta filosofía tiene el mismo fin que la que ofrece el lenguaje CSS en el caso del diseño web: separar el diseño del contenido.

Se ha optado por un estilo que enfrente tonos de negros y grises con tontos más claros de marrón o sepia. A su vez, se han personalizado los cuadros de los diálogos y elementos, como botones o cuadros de texto cuentan con marcos redondeados.

No obstante, llevar a cabo el diseño de la aplicación como se ha descrito anteriormente, permite que sea muy fácil modificarlo, pues bastaría con retocar dichos ficheros sin la necesidad de alterar el resto de la aplicación.

4.2.3 Base de datos local

El dispositivo móvil cuenta con una base de datos propia. Esta es gestionada por SQLite.

La base de datos local del móvil se utilizará para almacenar los mensajes ya leídos por el usuario en los distintos temas de las asignaturas. Esto permite realizar menos peticiones al servidor ya que solo es necesario solicitar los nuevos mensajes.

Por ejemplo, si un tema consta de 53 mensajes, una vez realizada la petición al servidor y plasmados en el dispositivo móvil, éste los almacena en la base de datos local junto a la fecha en la que se realizó dicha petición. Si pasado un tiempo, el usuario vuelve a abrir el tema y éste ya cuenta con 61 mensajes, la petición únicamente solicita los mensajes que están pendientes de leer desde el último acceso del usuario al tema. Por lo que, la aplicación tendría disponible instantáneamente los mensajes de la anterior entrada y el servidor nos devolvería los 8 mensajes nuevos.

4.3 Servidor

La API del servidor viene dada por la relación entre los ficheros PHP (API de acceso o controlador) y la base de datos principal (modelo). En este apartado se amplía la información.

4.3.1 Base de datos

La base de datos global se encuentra alojada en el servidor. En ella se almacena todo tipo de información: Usuarios, temas, asignaturas, mensajes, etc.

La ilustración 9 muestra el diagrama E-R en el que se basa dicha base de datos.

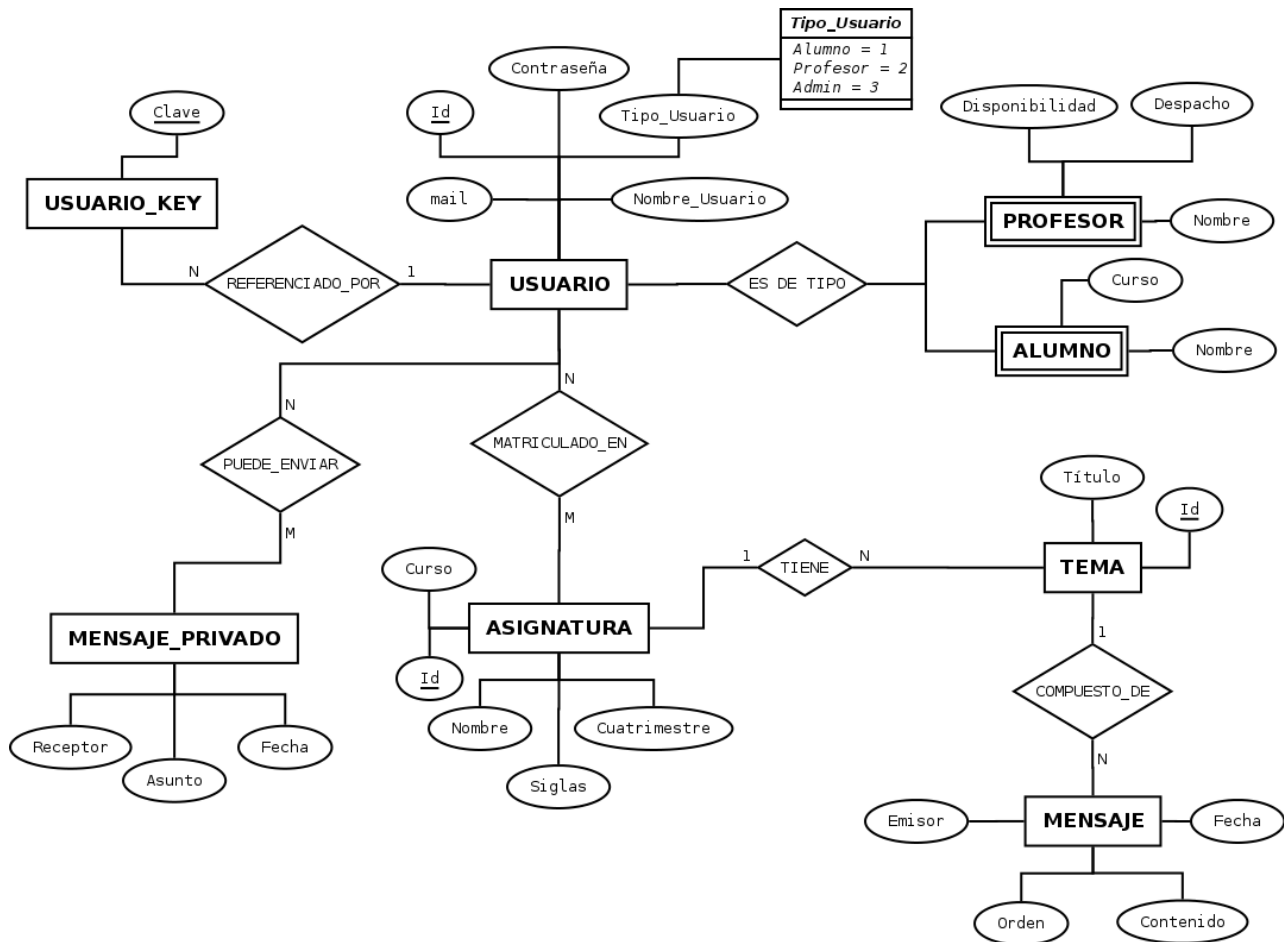


Ilustración 9: Diagrama Entidad-Relación base de datos servidor

4.3.2 PHP

La API de acceso se basa en ficheros PHP cuyas funciones nos permiten acceder a la base de datos principal para obtener o almacenar información.

En la tabla 1 se muestran los ficheros creados, indicando los parámetros que han de pasarse para su correcto uso y lo que retornan en base a ello.

Nombre del fichero	Parámetros que recibe	Devuelve
login	Nombre de usuario y contraseña	Permite o deniega la entrada de un usuario <ul style="list-style-type: none">o [tipoUsuario] [idUsuario] si el login es correctoo -1 si el login es incorrecto
getAsignaturas ByUsuario	Id del usuario	Devuelve las asignaturas de un usuario indicado <ul style="list-style-type: none">o Las asignaturas que cursa o imparte un usuario bajo el formato: [idAsignatura] [siglas] [curso] [cuatrimestre] [nombre]o -1 si el id el usuario no existeo -2 si el usuario no tiene asignaturas
getDatos ByUsuario	Id del usuario	Devuelve los datos de un usuario solicitado <ul style="list-style-type: none">o Si es profesor devuelve sus datos con el formato: [nombre] [email] [despacho] [disponibilidad]o Si es alumno devuelve los datos con el formato: [nombre] [email] [curso más alto matriculado]o -1 Si el usuario no existe
nuevoTema	Id de la asignatura y título del tema	Crea un nuevo tema en la asignatura indicada. <ul style="list-style-type: none">o Id del tema creadoo -1 Si no se introdujo alguno de los parámetros
borrarTema	Id del tema a eliminar	Elimina un tema y los mensajes que contuviera <ul style="list-style-type: none">o Id del temao -1 si no existe el tema indicado
getTemas ByAsignatura	Id de la asignatura	Devuelve los temas de una asignatura solicitada <ul style="list-style-type: none">o Temas que tiene la asignatura con el formato: [idTema] [tituloTema] [usuarioCreador] [fechaCreado] [numeroMsj] [fechaUltMsj]o -1 si no existe la asignaturao -2 si la asignatura no tiene temas vinculados
nuevoMensaje	Id del tema, emisor del mensaje y contenido del mensaje	Crea un nuevo mensaje en un tema indicado <ul style="list-style-type: none">o Fecha de la creación del mensajeo -1 si el id del tema o del emisor no existeno -2 no se rellenaron correctamente los parámetros
getMensajes ByTema	Id del tema	Devuelve los mensajes de un tema concreto <ul style="list-style-type: none">o Mensajes del tema con el formato: [idUsuarioEmisor] [nickUsuario] [contenidoMensaje] [fechaEnvioMensaje]o -1 si el tema no existe

nuevoMensaje Privado	Id del emisor, id del receptor y contenido del mensaje	<p>Crea un mensaje privado</p> <ul style="list-style-type: none"> ○ Lista de mensajes privados del usuario ○ -1 si no existe el id del usuario emisor o receptor ○ -2 no se rellenaron correctamente los parámetros
getMensajes Privados	Id del usuario	<p>Retorna los mensajes privados de un usuario.</p> <ul style="list-style-type: none"> ○ Devuelve los mensajes enviados y recibidos con el formato: [idEmisor] [idReceptor] [asunto] [fecha] [contenido] ○ -1 si el usuario no existe ○ -2 si el usuario no ha enviado o recibido mensajes
getProfesores ByAsignatura	Id de la asignatura	<p>Devuelve los profesores que imparten la asignatura</p> <ul style="list-style-type: none"> ○ Devuelven la lista de profesores con el formato: [idProfesor] [nombre] [email] [despacho] [disponibilidad] ○ -1 si la asignatura no existe
gcm	Tipo de notificación, emisor, asignatura, contenido...	<p>Gestiona todo lo relacionado con las notificaciones push: obtiene las claves de los usuarios a notificar y contacta con GCM para hacer llegar las notificaciones a los dispositivos correspondientes.</p>
registrarClave	Id del usuario y clave de GCM	<p>Vincula la clave recibida con el usuario indicado para tratar futuras notificaciones a dicho usuario.</p>

Tabla 1: Ficheros PHP servidor

Implementación

5.1 Tecnologías utilizadas

En este apartado se enumeran y explican las principales herramientas y tecnologías utilizadas durante el desarrollo del proyecto.

5.1.1 Eclipse

Eclipse es un entorno de desarrollo compuesto por un conjunto de herramientas de programación de código abierto multiplataforma. Eclipse permite añadir nuevas funcionalidades mediante *plugins*. En nuestro caso, para la codificación de aplicaciones Android, utilizaremos el *plugin* Android Development Tools (ADT) [10], que va a permitir desarrollar, depurar y emular de una manera eficiente y rápida el proyecto Android.

5.1.2 PHP

Las llamadas que se realizarán desde el dispositivo móvil al servidor serán a ficheros en lenguaje PHP. Este lenguaje es de código abierto y suele ser utilizado para el desarrollo web ya que permite ser incrustado en lenguaje HTML. También, como es nuestro caso, sirve para la gestión de peticiones http.

Utilizaremos los ficheros PHP para acceder a la base de datos y poder realizar consultas en función de los parámetros que pasemos a través de la URL por el método POST.

5.1.3 PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos relacionales de código abierto.

La base de datos, ubicada en el servidor, es gestionada por PostgreSQL. En ella se almacenan todos los datos a los que necesitamos acceder de forma remota a través de la aplicación.

5.1.4 Volley

Volley [11] es una librería desarrollada para optimizar el envío de peticiones http de servidores Android hacia servidores externos.

Hemos elegido Volley puesto que actúa como una interfaz de alto nivel, lo que nos permite liberarnos de la administración de hilos y procesos. Del mismo modo, ofrece disco transparente y almacenamiento en caché.

5.1.5 Imágenes 9-Path

Esta herramienta viene incluida en el SDK de Android. Es una clase que permite crear una imagen de bits dividida en nueve o más secciones. Se trata de marcar con unas barras solamente las zonas de la imagen que el sistema redimensionará para que, de esta forma, la imagen escale de la forma indicada cuando el contenido agregado dentro de esta imagen exceda los límites.

Ha resultado útil sobre todo para la adaptación óptima de los *bocadillos* de los chats en función de su contenido. En la ilustración 10 se puede apreciar en la figura izquierda como la zona verde es la que mantiene su grosor y márgenes laterales, mientras que la zona rosa es la que se expande o contrae en función de su contenido. Los bordes quedan intactos. A la derecha dos efectos visuales de cómo quedaría el contenedor.

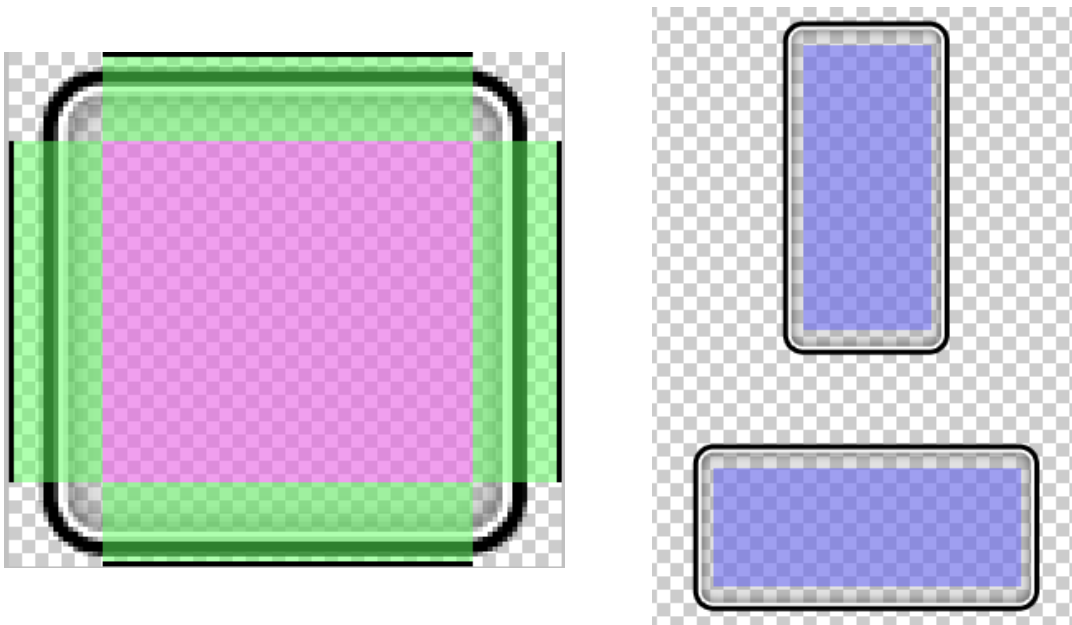


Ilustración 10: Imágenes Nine-Path

5.2 Google Cloud Messaging

GCM es un servicio de Google que permite a los desarrolladores enviar mensajes push a dispositivos móviles con diferentes sistemas operativos como Android o iOS.

La gran ventaja que nos ofrece la tecnología push es que es el servidor el encargado de ponerse en contacto con el dispositivo para informarle de actualizaciones. Esto evita que sea la propia aplicación la que pregunte al servidor constantemente obteniendo así un ahorro de energía en el dispositivo y una reducción en el tráfico de red.

La ilustración 11 muestra la arquitectura que ofrece GCM:

- La primera vez que el dispositivo móvil arranca la app, ésta lanza una petición a GCM para obtener un id que lo identifique (paso 1). GCM retorna al dispositivo un código alfanumérico que permite identificar de manera única al terminal (paso 2).
- A continuación, la app avisa al servidor propio indicándole el identificador y usuario al que estará asociada la nueva clave (paso 3).
- Se almacenan dichos datos en la base de datos alojada en el servidor con el fin de realizar futuras notificaciones (paso 4).
- En lo sucesivo, cuando sea necesario avisar a la app de alguna actualización, el servidor lanzará una petición (paso a) al servicio de GCM indicándole los identificadores de los usuarios a notificar, así como el contenido de la notificación. El servidor GCM enviará un mensaje push a los dispositivos correspondientes (paso b).

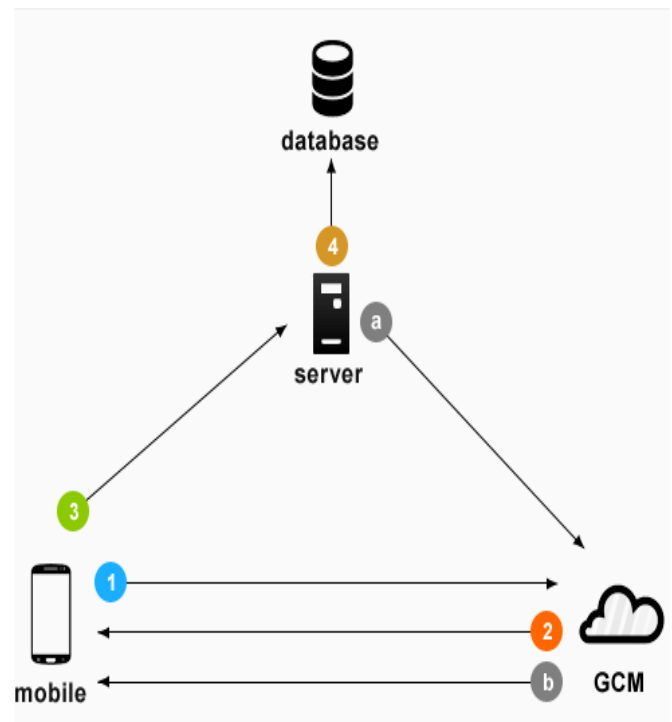


Ilustración 11: Arquitectura GCM [12]

5.3 Flujo de la aplicación

La ilustración 12 muestra la transición de pantallas que sigue PRALUAM en función de la interacción del usuario con la aplicación. En las siguientes secciones entraremos en detalle de cada pantalla y sus transiciones.

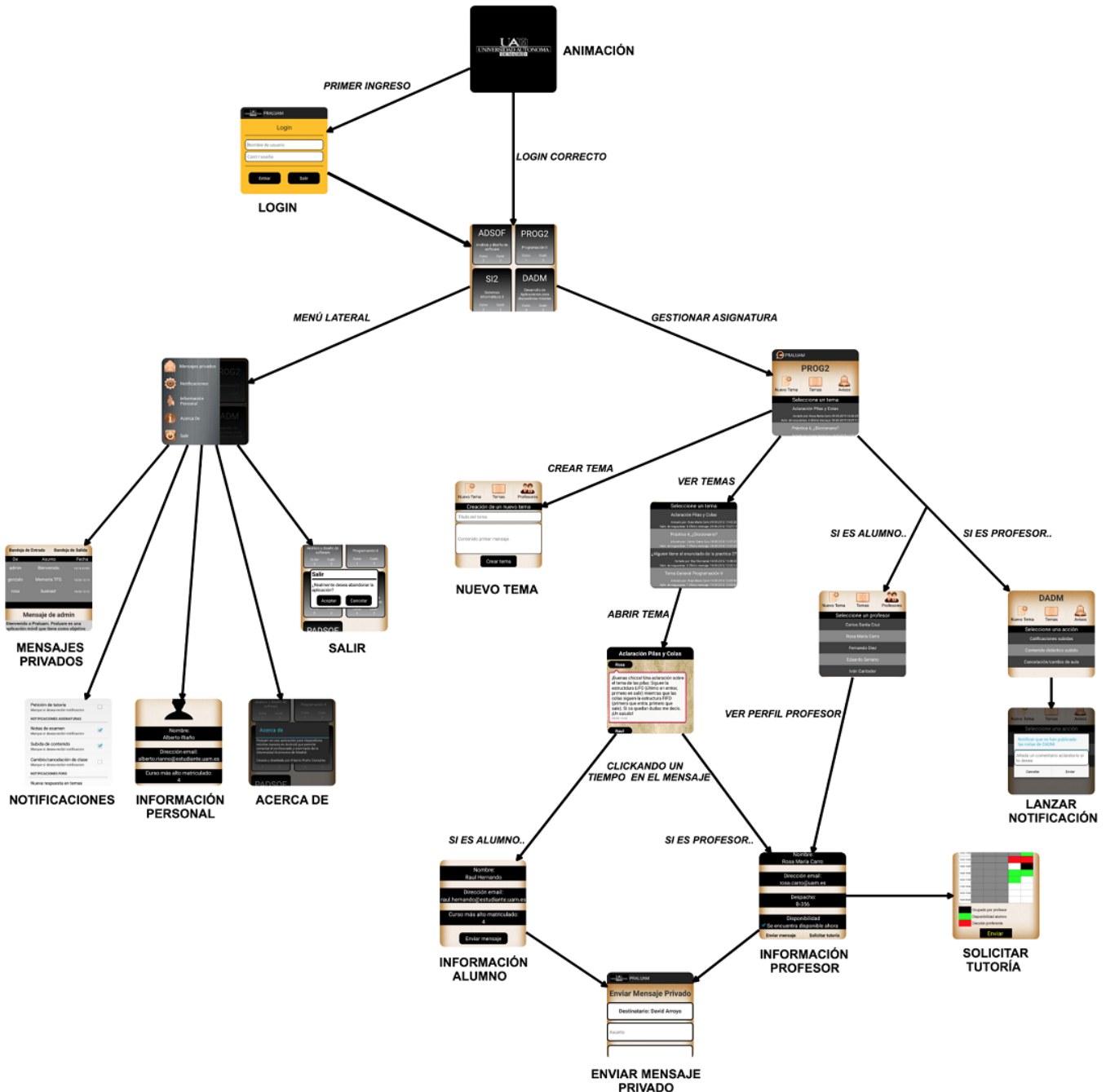


Ilustración 12: Diagrama de transición de pantalla de la aplicación

5.4 Componentes de la interfaz gráfica

En este apartado se muestra la implementación de la interfaz gráfica del proyecto y se indican las funcionalidades a las que responde cada pantalla.

5.4.1 Login y pantalla de inicio

La aplicación se abre con una breve animación que el usuario puede saltar si lo desea. En ese transcurso, la aplicación comprueba si el usuario ya se encuentra *logueado* para directamente dirigirle a la pantalla de inicio (Pantalla 2).

En caso contrario (Pantalla 1), el usuario tendrá que introducir sus credenciales para acceder a la aplicación. Si la combinación de usuario y contraseña resulta incorrecta o existe algún problema con la conexión de red se notifica al usuario. Por el contrario, se cargan las asignaturas que cursa/imparte el alumno/profesor y se abre la pantalla de inicio.



Ilustración 13: Conjunto de imágenes login y pantalla de inicio

La pantalla de inicio muestra las asignaturas del usuario. Pinchando en una de ellas la aplicación abrirá el menú de la asignatura que explicaremos en el apartado 5.2.3 Menú asignatura.

A su vez, bien clickando en el logotipo de la parte superior izquierda o arrastrando hacia dentro con el dedo, se muestra un menú desplegable (Pantalla 3) que ofrece diferentes opciones que analizamos en el siguiente punto.

5.4.2 Menú desplegable

Cuando el usuario despliegue este menú podrá tratar con 5 funcionalidades: envío de mensajería privada, configuración de notificaciones, información personal, visualización del *acerca de* y salir de la aplicación.

En primer lugar, la mensajería privada permite el intercambio de mensajes entre los usuarios de Praluam. Esta funcionalidad la explicaremos más detalladamente en el apartado 5.3.6 Mensajería privada.

Pinchando en notificaciones, la aplicación muestra la Pantalla 1 en la que el usuario podrá seleccionar de qué desea ser notificado. Para ello activará o desactivará mediante un checkbox las distintas opciones. En el caso de los temas del foro, podrá elegir entre: ninguno, los que ha iniciado, los que ha creado y todos.



Ilustración 14: Conjunto de imágenes menú desplegable

El apartado de información personal (Pantalla 2), únicamente mostrará los datos del usuario. Si el usuario en cuestión es profesor, le permitirá a su vez activar o desactivar el checkbox que hace referencia a su disponibilidad actual.

La pantalla de *Acerca de* (Pantalla 3) mostrará un breve mensaje resumiendo en pocas líneas la finalidad de la aplicación así como su versión.

Si el usuario pulsa en salir, el sistema lanzará un aviso solicitando confirmación. Si pulsa el botón de aceptar, la aplicación se cerrará y la siguiente apertura se llevará a cabo con la pantalla de *Login*.

5.4.3 Menú Asignatura

Al abrir el menú de una asignatura aparecen sus temas existentes. La interfaz aporta información sobre el título, quién y en qué fecha lo inicio, el número total de mensajes y cuándo fue la última respuesta. Los temas se ordenan a su vez por este último parámetro.



Ilustración 15: Conjunto de imágenes interacción con temas

El menú, a su vez, permite crear un tema (pantalla 3) y, en función de si el usuario es alumno o profesor (pantalla 1 ó 2) da opción a ver los profesores de dicha asignatura (pantallas 4 y 5) o lanzar un aviso a los alumnos (pantallas 6 y 7).



Ilustración 16: Conjunto de imágenes ver profesores y lanzar avisos

5.4.4 Interacción con temas de la asignatura

Cuando un usuario pulsa sobre un tema como los mostrados en la pantalla 1 se abre dicho tema. Se cargan los mensajes existentes en él (pantalla 2), situando a la derecha los enviados por el mismo y a la izquierda los ajenos. Si se pulsa de forma prolongada sobre alguno de estos se da la opción de visitar su perfil de usuario como muestra la pantalla 3.

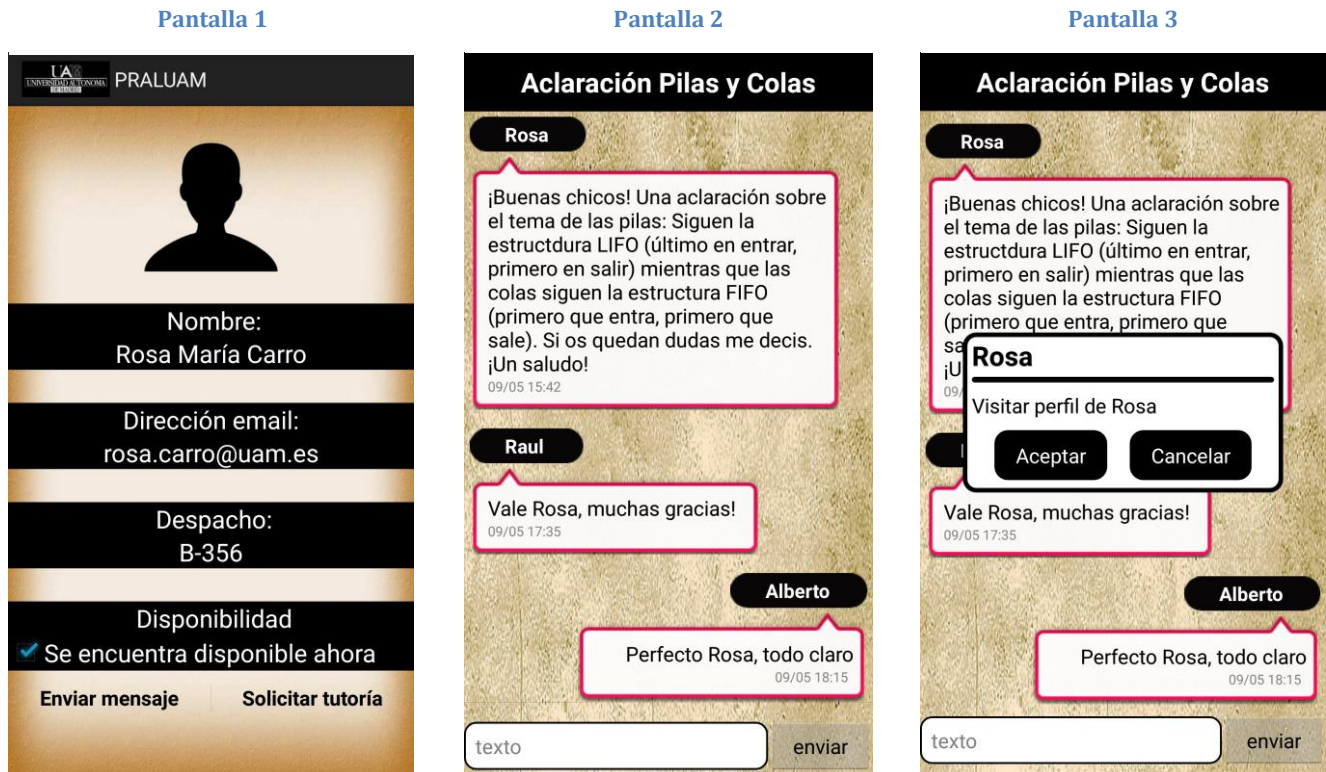


Ilustración 17: Conjunto de imágenes interacción con temas

El usuario puede escribir un nuevo mensaje en el tema a través del botón enviar. Si algún otro usuario tiene el tema abierto en ese momento verá reflejado el nuevo mensaje de forma instantánea. En caso contrario, si así lo tiene configurado, recibirá la notificación correspondiente como muestra la ilustración 18. Al abrir la notificación la aplicación se abre por el tema en cuestión.



Ilustración 18: Notificación nuevo mensaje en tema

5.4.5 Establecimiento de tutoría

Un alumno puede solicitar una tutoría visitando el perfil del profesor (pantalla 1). Para ello, rellena una solicitud indicando su disponibilidad. En ella podrá incluir algunas opciones como preferentes pulsando prolongadamente sobre la opción seleccionada (pantalla 2).

El profesor recibe la notificación (pantalla 5) y, de estar conforme, escoge la franja horaria a la que mejor se adapte (pantalla 3). Desde ese momento, se establece la tutoría avisando al alumno del día y hora escogidos por el profesor (pantalla 6). En ambos casos se da la opción de almacenar el evento en el calendario del móvil. Dicho evento servirá de recordatorio avisando a los implicados el día de la cita (pantalla 4).

Pantalla 1

Pantalla 2

Pantalla 3

Pantalla 4

Pantalla 5

Pantalla 6

Ilustración 19: Conjunto de imágenes establecimiento de tutoría

El profesor, si ninguna de las opciones le va bien, también tiene la opción de rechazar la tutoría pulsando el botón “Rechazar”. De ser así, se notifica al alumno tal como se muestra en la ilustración 20 ofertándole que realice una nueva petición

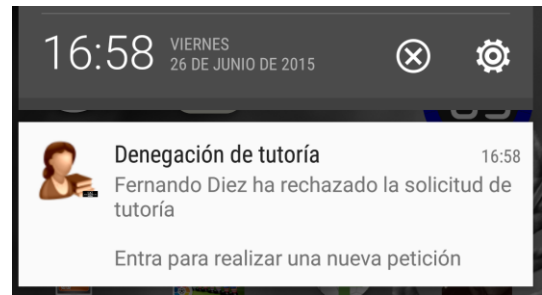


Ilustración 20: Denegación de tutoría

5.4.6 Mensajería privada

Praluam también ofrece un servicio de mensajería privada. Se puede enviar un mensaje a un destinatario seleccionando enviar mensaje tras visitar su perfil (pantalla 1).

Como vimos anteriormente, el menú desplegable daba acceso a las bandejas de mensajería (pantalla 2). En ella se agrupan los mensajes recibidos y enviados. Sin necesidad de salir de la pantalla, se puede responder a los mensajes recibidos (pantalla 3).

Pantalla 1

Pantalla 2

UNIVERSIDAD POLITÉCNICA DE MADRID

PRALUAM

Mensajes Privados

Bandeja de Entrada

Bandeja de Salida

De	Asunto	Fecha
admin	Bienvenido	13:14 07/05
gonzalo	Memoria TFG	18/06 12:15
gonzalo	Memoria TFG	18/06 12:15
rosa	buenas!	18/06 12:12

Mensaje de gonzalo

¿Te viene bien el lunes a las 12:00 en mi despacho?

Saludos, Gonzalo

Responder

Pantalla 3

Ilustración 21: Conjunto de imágenes mensajería privada

Si un usuario recibe un mensaje habiendo configurado ser avisado cuando esto suceda aparecerá en su dispositivo una notificación como la que muestra la Ilustración 21. En caso contrario, el usuario podrá ver el contenido del mensaje privado cuando acceda la próxima vez a la aplicación.

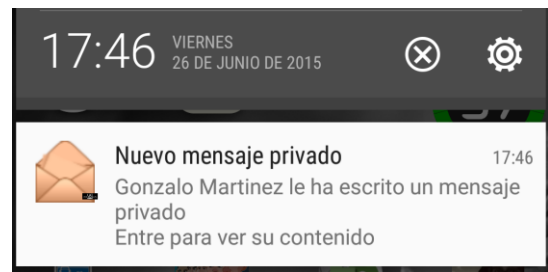


Ilustración 22: Notificación nuevo mensaje privado

Pruebas de la aplicación

Se deben separar las pruebas que agrupan la totalidad del proyecto. En primer lugar, se lleva a cabo la fase de pruebas de la parte cliente (la aplicación móvil), y en segundo lugar, la parte del servidor (PHP's).

6.1 Pruebas en el cliente

Se han agrupado las pruebas según las principales funcionalidades que ofrece la aplicación PRALUAM. En cada prueba se detalla qué requisito o requisitos atiende a partir de los redactados en el apartado 3.1.

A su vez, se establecen 4 tipos diferentes de pruebas:

Tipo 1 Funcionalidad correcta: Se comprueba si las principales funcionalidades que ofrece Pralum responden de la forma esperada.

Tipo 2 Control de errores: Se ha de notificar al usuario sobre aquellas acciones que puede llevar a cabo pero para las que no está diseñada la aplicación.

Tipo 3 Casos límite: La aplicación ha de reaccionar bien cuando el usuario lleva a cabo acciones anómalas pero que se encuentran dentro de lo permitido por la aplicación.

Tipo 4 Visualización: La interfaz gráfica se debe adaptar de forma correcta para todo tipo de situaciones extremas y dispositivos que cumplan las condiciones.

Inicio y cierre de la sesión:

Tipo de Prueba	Requisitos a los que corresponde	Prueba realizada	Resultado
1	RF1, RNF3	Logueo en la aplicación introduciendo correctamente usuario y contraseña	Se redirige al usuario al menú principal con sus asignaturas correctamente cargadas
2	RF1, RNF3	El usuario no introduce nombre de usuario y/o contraseña en el momento de intentar iniciar sesión	La aplicación responde con un mensaje de alerta solicitando al usuario que rellene ambos campos
2	RF1, RNF1, RNF3	El dispositivo móvil del usuario se encuentra sin conexión a internet	Se notifica al usuario que existe un error de conexión y que la aplicación no puede funcionar sin conexión a internet.

2	RF1, RNF3, RNF4	La petición al servidor tarda demasiado en responder por la mala conexión del dispositivo	Mensaje de alerta indicando al usuario la mala conexión actual y solicitando que lo intente de nuevo
1	RF2, RNF3	El usuario sale de la aplicación sin cerrar sesión	La siguiente entrada a la aplicación por parte del usuario mantendrá su sesión y será redirigido al menú principal
1	RF2, RNF3	El usuario cierra la sesión desde la aplicación	Se solicita confirmación del cierre y, de aceptarse, la siguiente entrada en la aplicación será a la pantalla de login

Tabla 2: Pruebas cliente: Inicio y cierre sesión

Interacción con los temas de la Asignatura:

Tipo de Prueba	Requisitos a los que corresponde	Prueba realizada	Resultado
1	RF5.2, RNF1	El usuario selecciona una de las asignaturas que cursa o imparte	La aplicación muestra los temas de dicha asignatura ordenados por la fecha de su última respuesta
1	RF5.1	El usuario crea un tema introduciendo título y contenido del primer mensaje	Se indica que el tema se ha creado de forma correcta y se añade a la lista de temas posicionándolo en primer lugar
2	RF5.1	El usuario no introduce título y/o contenido del primer mensaje cuando intenta crear un tema	La aplicación responde con un mensaje de alerta solicitando al usuario que rellene ambos campos
2	RF5.1, RNF4	El usuario introduce un título excesivamente largo (>50 caracteres)	El sistema avisa de que la longitud máxima para el título de un tema es de 50 caracteres
3	RF5.1, RNF2	Se introduce un primer mensaje tan largo que no cabe en las 5 líneas que ocupa en un primer lugar el editText que lo almacena	El editText se adapta y va alargándose creando líneas adicionales. En ningún momento se deja de ver lo que el usuario escribe
1	RF5.2, RNF1, RNF4	El usuario selecciona un tema	Se cargan los mensajes de dicho tema. En el caso de que lo hubiera abierto con anterioridad, los mensajes ya leídos aparecerán de forma instantánea y los nuevos aparecerán en menos de 1seg
4	RF5.2, RNF1, RNF2	El usuario selecciona un tema	Se visualizan todos los mensajes correctamente mediante un scroll si fuera necesario. Se posiciona visualmente sobre el último mensaje

1	RF3, RF5.2, RNF1	Un usuario escribe un nuevo mensaje en el tema.	Se notifica a todos los usuarios que así lo tengan configurado del nuevo mensaje. Si alguno de estos usuarios tuviera abierto el tema en ese momento, apreciará de forma instantánea el nuevo mensaje en su dispositivo
2	RF5.2, RNF4	El usuario intenta enviar un mensaje vacío al tema	No se crea un nuevo mensaje.
4	RF5.2, RNF2	Se escribe en el tema mensajes de variadas longitudes	Los bocadillos se adaptan de forma correcta manteniendo siempre los bordes intactos y unos márgenes determinados para su correcta visualización

Tabla 3: Pruebas cliente: Interacción con temas de asignaturas

Intercambio de Mensajería Privada:

Tipo de Prueba	Requisitos a los que corresponde	Prueba realizada	Resultado
1	RF3, RF4, RNF4	El usuario envía un mensaje privado tras visitar el perfil de usuario del destinatario	Se da una confirmación al usuario indicándole que el mensaje se ha enviado correctamente. Al receptor le llega una notificación sobre el nuevo mensaje privado que, de clickarla, la aplicación se abre en la bandeja de entrada y permite contestar desde la misma pantalla
2	RF4	El usuario no introduce el campo Asunto en el mensaje	La aplicación pregunta si se desea enviar el mensaje privado sin asunto
2	RF4, RNF4	El usuario no rellena el contenido del mensaje a enviar	El sistema avisa de que el contenido del mensaje no puede quedar vacío
4	RF4, RNF2	El número de mensajes no cabe en la tabla destinada a la bandeja de entrada o salida	Un scroll que sólo ocupa dicha tabla permite visualizar todos desplazando el dedo

Tabla 4: Pruebas cliente: Intercambio de mensajería privada

Establecimiento de tutoría y avisos sobre asignaturas:

Tipo de Prueba	Requisitos a los que corresponde	Prueba realizada	Resultado
1	RF6	El alumno envía una solicitud de tutoría al profesor correspondiente	El profesor recibe correctamente el aviso y puede fijar la tutoría seleccionando alguna de las propuestas enviadas por el alumno. También puede rechazarla. En ambos casos se notifica al alumno con la decisión final
2	RF6, RNF3	El alumno no rellena ninguna de las opciones para la solicitud de tutoría	La aplicación informa de que debe seleccionarse al menos una opción
2	RF6	El profesor no puede establecer una tutoría en las opciones que el alumno propuso	La aplicación permite al profesor rechazar la tutoría tras solicitar confirmación. El sistema notifica al alumno de que el profesor ha rechazado su solicitud y le da opción a iniciar una nueva con diferentes fechas
3	RF6, RNF1, RNF4	Se envía una solicitud pero el dispositivo móvil del receptor está apagado o sin conexión a internet en ese momento	La notificación aparece en el dispositivo a los pocos segundos de encenderse o volver a tener conexión a internet
4	RF6, RNF2	Al solicitar una tutoría, comprobar que los cuadrantes seleccionados corresponden a las opciones de horas y días deseados por parte del alumno	La selección se lleva a cabo correctamente

Tabla 5: Pruebas cliente: Tutoría y avisos de asignaturas

6.2 Pruebas en el servidor

La parte del servidor se codificó bajo el sistema operativo Ubuntu, utilizando Apache y trabajando en *localhost*. Los errores en los ficheros PHP que se iban produciendo han sido localizados revisando el fichero *error.log* situado en el directorio */var/log/apache2/*.

Se ha comprobado exitosamente el funcionamiento de los ficheros enumerados en el apartado 4.3.2 *PHP* cuando los parámetros son correctamente introducidos. Las pruebas teniendo en cuenta los casos límite se encuentran redactadas en la tabla 6.

Fichero en el que se realiza la prueba	Prueba realizada	Resultado esperado	Resultado obtenido
login.php	Se introduce un nombre de usuario y contraseña erróneamente.	-1	-1
getAsignaturasBy Usuario.php	Se introduce un id de un usuario no almacenado en la base de datos	-1	-1
getAsignaturasBy Usuario.php	El usuario no tiene asignaturas asociadas	-2	-2
getDatosBy Usuario.php	Se introduce un id de un usuario no almacenado en la base de datos	-1	-1
nuevoTema.php	No se introduce título en el tema a crear	-1	-1
borrarTema.php	Se borra un tema comprobando que los mensajes que dependían de él también son borrados	OK	OK
getTemasBy Asignatura.php	El id introducido no se corresponde con ninguna asignatura.	-1	-1
nuevoMensaje.php	Se deja en blanco el contenido del mensaje a enviar	-2	-2
getMensajesBy Tema.php	El id introducido no se corresponde con ningún tema almacenado en la base de datos.	-1	-1
getMensajes Privados.php	La solicitud se hace sobre un usuario que aún no ha recibido ningún mensaje privado	-2	-2
getProfesores ByAsignatura.php	El id de la asignatura sobre la que se realiza la petición no existe	-1	-1
gcm.php	El campo <i>registration_ids</i> está vacío porque los usuarios a los que se debía avisar no tienen clave asociada.	-1	-1
registrarClave.php	Se registra una clave a un usuario que ya contaba con una clave	OK, el usuario tiene dos dispositivos asociados	OK

Tabla 6: Pruebas servidor

Conclusiones y trabajo futuro

En este apartado se precisan las conclusiones personales obtenidas a partir del desarrollo del proyecto y posibles modificaciones del mismo para su mejora.

7.1 Conclusiones

En primer lugar, todos los objetivos y requisitos mencionados en los capítulos anteriores se han llevado a cabo. El objetivo principal del proyecto era el diseño, implementación y evaluación de una aplicación para dispositivos móviles que posibilitará una mayor y mejor conexión entre alumnos y profesores.

La forma de aplicar dicho objetivo es ofertando, a través de la aplicación, una serie de funcionalidades principales: intercambio de mensajes privados, establecimiento de tutorías, envío de notificaciones sobre novedades en las asignaturas y, principalmente, un servicio de mensajería instantánea para tratar temas relativos a las asignaturas.

Atendiendo a la parte técnica del proyecto, ha resultado especialmente beneficioso dedicar suficiente tiempo a sintetizar todos los requisitos que iba a ofertar la aplicación. Una vez realizadas dichas fases de análisis y diseño permitió que las fases posteriores pudieran ir más fluidas y con menos obstáculos de lo previsto.

Según iba desarrollando el proyecto, y aprovechaba para mostrar el funcionamiento y variantes que ofrecía la aplicación tanto a amigos como familiares, me he ido convenciendo, a partir de sus impresiones, de que realmente podría llegar a ser muy beneficioso que las universidades contasen con una aplicación de estas características.

A nivel personal, me ha servido para sentirme capaz de llevar a cabo un proyecto de principio a fin, pasando por todas sus fases. También he adquirido nuevos conocimientos, o ampliado muchos de ellos en materias como Java, PHP, SQL, GCM.

7.2 Trabajo futuro

El proyecto es susceptible de albergar múltiples mejoras. La aplicación está diseñada para facilitar la incorporación de nuevos módulos, ya sean con el fin de mejorar sus funcionalidades actuales como para añadir nuevas.

El objetivo futuro de este proyecto sería brindar a todos los usuarios la posibilidad de tener todo lo relacionado con la universidad en una única aplicación móvil.

En esta sección se comentarán posibles modificaciones que permitirían lo ya comentado.

7.2.1 Vinculación directa con Moodle o Sigma

El adquirir derechos para llevar a cabo una asociación entre la aplicación y plataformas educativas como Moodle o Sigma posibilitaría ventajas como:

- Hacer el login contra Moodle lo que permitiría realizar la carga inicial de asignaturas en las que los alumnos y profesores están involucrados.
- Profesores y alumnos podrían acceder desde la misma aplicación directamente al contenido que ofrece Moodle. Esto permitiría al alumno ver las notas de la asignatura en el momento que reciba la notificación de que éstas han sido publicadas. También ofrecería una mayor comodidad a la hora de ver un pdf concreto con el fin de resolver una duda.
- El alumno podría ver el expediente completo, consultando sus calificaciones a lo largo de la carrera, su nota media, el cierre de actas, etc.
- Al finalizar las asignaturas se llevan a cabo una serie de encuestas opcionales para valorar, de forma anónima, la forma de impartir la asignatura por parte del profesorado. La aplicación podría albergar la opción de realizar la encuesta desde el propio dispositivo ofertando mayor comodidad así como la opción de que los alumnos fueran avisados mediante una notificación de la disponibilidad de realizar dichas encuestas.

7.2.2 Calendario personal para el alumno

Una vez cargado el perfil del alumno, conociendo sus asignaturas, horarios y grupos en los que se encuentra matriculado ofertarle la opción de ver un calendario personificado en el que podrá:

- Ver en qué día y hora tiene sus clases. En qué aula (o laboratorio en el caso de ser una clase práctica) se dan o qué profesor las imparte.
- Visualizar de manera resaltada los días en los que haya programado un examen de una de las asignaturas que curse el alumno.
- Añadir notas en las asignaturas o fechas como por ejemplo indicando quién es tu compañero en las prácticas o apuntando qué conviene tener estudiado para el siguiente día de clase.

7.2.3 Relación con servicios internos de la Universidad

La idea principal de poder tener en una única aplicación todo lo relacionado con la universidad nos podría llegar a ofrecer otras vinculaciones con servicios como:

- **Biblioteca:** Los estudiantes podrían consultar desde el mismo dispositivo móvil la facultad donde se encuentra un libro y la disponibilidad del mismo, así como la posibilidad de realizar una reserva en el caso de éste no estar disponible. También se podría renovar el préstamo si no existe lista de espera o consultar el tiempo de sanción restante en el supuesto de que el usuario no entregara en plazo un libro.
- **Servicio de deportes:** Los alumnos podrían realizar consultas sobre ofertas, reservas de instalaciones deportivas, clasificaciones de las competiciones internas de la universidad, etc.
- **Cafeterías de las facultades:** Los usuarios podrían consultar productos y precios que ofrece cada una de las cafeterías. Podría ser de utilidad a la hora de decidir en qué facultad comer acorde a los menús del día que oferta cada una de las cafeterías.

Referencias

- [1] <http://www.ine.es/prensa/np864.pdf>
- [2] [http://www.htcmania.com/mediawiki/index.php/Programaci3n_de_aplicaciones_para_m3viles_Android - Unidad 1](http://www.htcmania.com/mediawiki/index.php/Programaci3n_de_aplicaciones_para_m3viles_Android_-_Unidad_1)
- [3] <https://es.wikipedia.org/wiki/Android>
- [4] <http://eduardotortosamartin.com/todas-las-versiones-de-android/>
- [5] <http://andro4all.com/2015/04/distribucion-android-marzo-2015>
- [6] <http://www.uc3m.es/Inicio>
- [7] <https://www.whatsapp.com>
- [8] <http://www.md5.net>
- [9] <https://developers.google.com/cloud-messaging/>
- [10] <http://developer.android.com/tools/sdk/eclipse-adt.html>
- [11] <https://developer.android.com/training/volley/index.html>
- [12] <https://www.androidhive.info/2012/10/android-push-notifications-using-google-cloud-messaging-gcm-php-and-mysql/>